

# TEXTURE MAPPING METHOD AND DEVICE THEREFOR

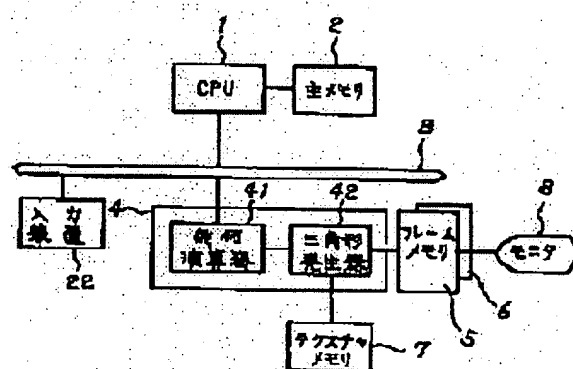
**Patent number:** JP7254072  
**Publication date:** 1995-10-03  
**Inventor:** SAKAIHARA TOORU; others: 02  
**Applicant:** HITACHI LTD  
**Classification:**  
 - **International:** G06T15/00  
 - **European:**  
**Application number:** JP19950009533 19950125  
**Priority number(s):**

Report a data error here

## Abstract of JP7254072

**PURPOSE:** To provide texture mapping method and device capable of easily expressing patterns, material feeling and ruggedness in animation display by simple hardware constitution.

**CONSTITUTION:** In a texture memory 7, plural texture data TX1-TX4 in the case of viewing a display object surface from some assumed directions are stored. When information for specifying the surface shape, the direction for viewing the surface and texture in the texture memory 7 is supplied as a command from a CPU 1 to a graphic processor 4, a geometrical operation device 41 forms the read information of the texture memory 7 from the information. A triangle generator 42 reads the texture data corresponding to the direction for viewing the display object surface by the read information from the texture memory 7 and performs texture mapping. The picture of the object surface for which the texture mapping is performed is stored in a frame memory 5 and displayed at a monitor 8.



Data supplied from the esp@cenet database - Patent Abstracts of Japan

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-254072

(43) 公開日 平成7年(1995)10月3日

(51) Int. Cl. 6	識別記号	庁内整理番号	F I	技術表示箇所
G06T 15/00		9365-5L	G06F 15/72	450 A

審査請求 未請求 請求項の数14 O L (全12頁)

(21) 出願番号 特願平7-9533  
(22) 出願日 平成7年(1995)1月25日  
(31) 優先権主張番号 特願平6-7017  
(32) 優先日 平6(1994)1月26日  
(33) 優先権主張国 日本 (J P)

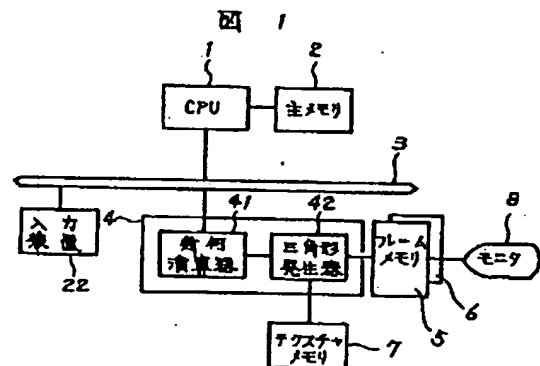
(71) 出願人 000005108  
株式会社日立製作所  
東京都千代田区神田駿河台四丁目6番地  
(72) 発明者 酒井原 徹  
神奈川県川崎市麻生区王禅寺1099番地株式  
会社日立製作所システム開発研究所内  
(72) 発明者 川崎 敏治  
神奈川県川崎市麻生区王禅寺1099番地株式  
会社日立製作所システム開発研究所内  
(72) 発明者 桑名 利幸  
茨城県日立市大みか町五丁目2番1号株式  
会社日立製作所大みか工場内  
(74) 代理人 弁理士 小川 勝男

(54) 【発明の名称】 テクスチャマッピング方法及び装置

(57) 【要約】

【目的】 簡単なハードウェア構成で、動画表示において、紋様、材質感、凹凸感を容易に表現可能なテクスチャマッピング方法及び装置を提供する。

【構成】 テクスチャメモリ7には、表示物体表面をいくつかの想定される方向から見た場合の複数のテクスチャデータ (TX1~TX4) を格納する。CPU1からグラフィックプロセッサ4にこの表面形状やこの表面を見る方向、テクスチャメモリ7でのテクスチャなどを指定する情報がコマンドとして供給されると、幾何演算器41はかかる情報からテクスチャメモリ7の読出し情報を形成する。三角形発生器42はこの読出し情報によって表示物体表面の見る方向に対応したテクスチャデータをテクスチャメモリ7から読出し、テクスチャマッピングする。テクスチャマッピングされた物体表面の画像はフレームメモリ5に記憶され、モニター8で表示される。



## 【特許請求の範囲】

【請求項1】表示されるべき物体の表面に関する形状の幾何学的情報および該表面の模様を指定するテクスチャデータと、表示される状況を設定するために仮想的に与えられる該物体に対する視点の情報とをもとに座標変換を施し、該テクスチャデータを表示されるべき該表面にマッピングすることにより、該表面の細部を表現する三次元図形表示において、

テクスチャデータのもとになる物体あるいは仮想的に生成される物体の表面の向きを予め種々想定し、想定される該向き毎にテクスチャデータを設定し、表示される該物体の表面の実際の向きに対応する該想定される向きのテクスチャデータを選択し、選択された該テクスチャデータを該表面にマッピングすることを特徴とするテクスチャマッピング方法。

【請求項2】請求項1記載のテクスチャマッピング方法において、

表示される前記物体表面の実際の向きに対応したテクスチャデータがない場合には、該実際の向きに最も近い複数の前記想定される向きのテクスチャデータを選択し、該実際の向きとこれに最も近い該複数の向きとの偏差に応じて選択された該テクスチャデータを補間処理することにより該実際の向きに対するテクスチャデータを求め、得られた該テクスチャデータを前記表面にマッピングすることを特徴とするテクスチャマッピング方法。

【請求項3】表示されるべき物体の表面に関する形状の幾何学的情報および該表面の模様を指定するテクスチャデータと、表示される状況を設定するために仮想的に与えられる該物体に対する視点の情報とをもとに座標変換を施し、該テクスチャデータを表示される該表面にマッピングすることにより、該表面の細部を表現する三次元図形表示において、

該表示される物体表面の向きが種々想定され、該向き毎に設定されるテクスチャデータを保持する記憶装置と、該物体の表示時、表示される該表面の実際の向きに対応した該想定される向きのテクスチャデータを該記憶装置から選択する選択機構と、

該選択機構によって選択された該テクスチャデータを表示される該表面にマッピングした画像を生成する図形生成手段とを含むことを特徴とするテクスチャマッピング装置。

【請求項4】請求項3記載のテクスチャマッピング装置において、

表示されるべき前記物体の表面の実際の向きに対応したテクスチャデータがない場合には、該実際の向きに最も近い複数の前記想定される向きのテクスチャデータを選択し、該実際の向きとこれに最も近い該複数の向きとの偏差に応じて選択された該テクスチャデータを補間処理し、これにより、該実際の向きに対するテクスチャデータを求める補間装置を設け、

該補間装置によって得られた該テクスチャデータを前記表面にマッピングすることを特徴とするテクスチャマッピング装置。

【請求項5】表示されるべき物体の表面に関する形状の幾何学的情報および該表面の模様を指定するテクスチャデータと、表示される状況を設定するために仮想的に与えられる該物体に対する視点の情報とをもとに座標変換を施し、該テクスチャデータを表示される該表面にマッピングすることにより、該表面の細部を表現する三次元図形表示において、

複数のテクスチャデータとこれらを補間する補間係数の指定を受け、これらに基づく補間処理を実行して実際の向きに対するテクスチャデータを得、得られた該テクスチャデータを該表面にマッピングした画像を生成する図形生成手段を含むことを特徴とするテクスチャマッピング装置。

【請求項6】表示すべき物体の表面に関する形状の幾何学的情報および該表面の模様を決定するテクスチャデータを、表示装置の表示画面上での該物体の姿勢に適合させて該表示画面上の該物体の占める領域にマッピングすることにより、該物体の表面の細部を表現するテクスチャマッピング方法であって、

複数の所定方向から見た前記物体の表面に関する複数のテクスチャデータを予めテクスチャメモリに記憶し、中央処理ユニットに入力される情報に基づいて、前記表示画面上の前記物体の姿勢を決定し、該姿勢を示すコマンドを該中央処理ユニットからグラフィックプロセッサへ送り、

前記グラフィックプロセッサにおいて、受け取ったコマンドに基づいて、前記テクスチャメモリに記憶されたテクスチャデータの中から適切な少なくとも1つのテクスチャデータを選択し、これを用いて前記表示画面上の前記物体の占める領域にマッピングすることを特徴とするテクスチャマッピング方法。

【請求項7】請求項6記載のテクスチャマッピング方法において、

前記グラフィックプロセッサは、前記物体の姿勢を表わすディスプレイメントの値とそれに対応する1つのテクスチャデータとの対応関係を記憶するテクスチャデータ管理テーブルを含み、

前記マッピングするステップにおいて、前記グラフィックプロセッサは前記テクスチャデータ管理テーブルにディスプレイメント値を与えることにより、1つのテクスチャデータを読み出し、これを用いて前記表示画面上の前記物体の占める領域にマッピングすることを特徴とするテクスチャマッピング方法。

【請求項8】請求項6記載のテクスチャマッピング方法において、

前記グラフィックプロセッサは、前記物体の姿勢を表わすディスプレイメントの値とそれに対応する複数のテク

スチャデータとの対応関係を記憶するテクスチャデータ管理テーブルを含み、

前記マッピングするステップにおいて、前記グラフィックプロセッサは前記テクスチャデータ管理テーブルにディスプレイメント値を与えることにより、複数のテクスチャデータを読み出し、該読み出したテクスチャデータに補間処理を行って新たなテクスチャデータを得て、これを用いて前記表示画面上の前記物体の占める領域にマッピングすることを特徴とするテクスチャマッピング方法。

【請求項9】請求項8記載のテクスチャマッピング方法において、

前記テクスチャデータ管理テーブルは、前記物体の姿勢を表わすディスプレイメントの値とそれに対応する複数のテクスチャデータと該複数のテクスチャデータに対する補間係数との対応関係を記憶し、

前記マッピングするステップにおいて、前記グラフィックプロセッサは、前記テクスチャデータ管理テーブルから読み出した複数のテクスチャデータに対して、読み出した補間係数を用いて補間処理を行うことを特徴とするテクスチャマッピング方法。

【請求項10】表示すべき物体の表面に関する幾何学的情報および該表面の紋様を決定するテクスチャデータを、表示装置の表示画面上での該物体の姿勢に適合させて該表示画面上の該物体の占める領域にマッピングすることにより、該物体の表面の細部を表現するテクスチャマッピング装置であって、

複数の所定方向から見た前記物体の表面に関する複数のテクスチャデータを予め記憶しておくテクスチャメモリと、

入力される情報に基づいて、前記表示画面上の前記物体の姿勢を決定し、該姿勢を示すコマンドを発行する中央処理ユニットと、

前記中央処理ユニットに接続され、受け取ったコマンドにตอบสนองして、前記テクスチャメモリに記憶されたテクスチャデータの中から適切な少なくとも1つのテクスチャデータを選択し、これを用いて前記表示画面上の前記物体の占める領域にマッピングするグラフィックプロセッサとを備えたことを特徴とするテクスチャマッピング装置。

【請求項11】請求項10記載のテクスチャマッピング装置において、

前記グラフィックプロセッサは、前記物体の姿勢を表わすディスプレイメントの値とそれに対応する1つのテクスチャデータとの対応関係を記憶するテクスチャデータ管理テーブルを含み、

前記グラフィックプロセッサは、前記テクスチャデータ管理テーブルにディスプレイメント値を与えることにより、1つのテクスチャデータを読み出し、これを用いて前記表示画面上の前記物体の占める領域にマッピングす

ることを特徴とするテクスチャマッピング装置。

【請求項12】請求項10記載のテクスチャマッピング装置において、

前記グラフィックプロセッサは、前記物体の姿勢を表わすディスプレイメントの値とそれに対応する複数のテクスチャデータとの対応関係を記憶するテクスチャデータ管理テーブルを含み、

前記グラフィックプロセッサは、前記テクスチャデータ管理テーブルにディスプレイメント値を与えることにより、複数のテクスチャデータを読み出し、該読み出したテクスチャデータに補間処理を行って新たなテクスチャデータを得て、これを用いて前記表示画面上の前記物体の占める領域にマッピングすることを特徴とするテクスチャマッピング装置。

【請求項13】請求項12記載のテクスチャマッピング装置において、

前記テクスチャデータ管理テーブルは、前記物体の姿勢を表わすディスプレイメント値とそれに対応する複数のテクスチャデータと該複数のテクスチャデータに対する補間係数との対応関係を記憶し、

前記グラフィックプロセッサは、前記テクスチャデータ管理テーブルから読み出した複数のテクスチャデータに対して、読み出した補間係数を用いて補間処理を行うことを特徴とするテクスチャマッピング装置。

【請求項14】表示画面上に景観を表示する表示装置と、

操作者による模擬的なハンドル操作、ブレーキ操作およびアクセル操作を少なくとも表わす入力情報を受け、該入力情報を示す信号を発生する入力装置と、

入力装置からの入力情報に基づいて、前記表示画面上に表示された景観中の物体の姿勢を決定し、該姿勢を示すコマンドを発行する中央処理ユニットと、

前記表示画面上に表示すべき前記物体を複数の所定方向から見た場合の表面に関する幾何学的情報および該表面の紋様を決定する複数のテクスチャデータを予め記憶しておくテクスチャメモリと、

前記中央処理ユニットに接続され、受け取ったコマンドにตอบสนองして、前記テクスチャメモリに記憶されたテクスチャデータの中から適切な少なくとも1つのテクスチャデータを選択し、これを用いて前記表示画面上の前記物体の占める領域にマッピングするグラフィックプロセッサとを備えたことを特徴とするドライビングシミュレータ。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、三次元図形の表示に係り、特に、紋様や材質感に加えて凹凸感を伴うリアルな物体表面の表示を比較的小規模のハードウェアで高速に行なうことができるテクスチャマッピング方法及び装置に関する。

## 【0002】

【従来の技術】特開平3-138779号公報に記載されるように、従来から計算機支援設計における意匠上の効果の確認などのためにテクスチャマッピングが用いられている。テクスチャマッピングとは、図形を細部まで精密に定義して表示する代わりに、表示物体の表面を比較的簡単な近似多面体で近似し、この表示物体の表面に紋様や材質、凹凸などを表わす画像（これをテクスチャあるいはテクスチャマップという）を貼り付る（即ち、マッピングする）ことにより、表示物体の表面に紋様や材質感あるいは凹凸感を表わすようにした技術である。

【0003】この方法では、表示物体を近似多面体で表わすようにしているが、かかる多角形の表面にテクスチャマッピングする場合には、次のような処理が行なわれる。まず、この多角形が光源によって照明されているものとして、この多角形の各頂点の輝度を、予め設定されている光源の位置と強度、物体の材質（従って、光の反射係数）及び頂点の法線から計算して求める。また、多角形内の各画素の輝度は、求められた頂点の輝度値から補間より求める。このようにして、多角形の各点での輝度が求められる。

【0004】一方、テクスチャは濃淡や色によって表わされる絵柄であって、このようなテクスチャの各点の濃さや色は数値でもって対応づけられている。この数値をテクスチャ値という。また、テクスチャはテクスチャ座標系と称する座標系上で定義されており、テクスチャ内の各位置はこのテクスチャ座標系での座標位置と一対一に対応している。そして、このテクスチャ座標系での座標位置には、この座標位置に対応するテクスチャの位置での濃さや色の数値、即ち、テクスチャ値が記憶されている。かかるテクスチャ値は、テクスチャデータとして、アドレスがテクスチャ座標系での座標位置に対応させたメモリに格納される。このようにして、各種のテクスチャをメモリに格納することができる。

【0005】多角形の表面にテクスチャをマッピングするためには、まず、多角形の各頂点に対して、マッピングすべきテクスチャのテクスチャ座標系での座標位置が設定される。つまり、テクスチャ座標系でのどの座標位置を多角形の頂点に対応させるか決める。このように、多角形の各頂点に対するテクスチャ座標系の座標位置が決まると、多角形内の各画素に対するテクスチャ座標系での座標位置が上記の求められた頂点の座標位置から補間によって求められる。そして、このようにして得られた多角形の頂点を含む各点の座標位置を用いてメモリからテクスチャデータがアクセスされ、テクスチャデータは上記のようにして得られた対応する輝度値と混合されてフレームメモリに書き込まれる。かかるフレームメモリのデータによってテクスチャマッピングされた多角形が表示されることになる。

【0006】また、特に物体の凹凸感を表現するため

に、従来、例えば白田耕作監訳「最新3次元コンピュータグラフィックス」（1991年 アスキー出版局刊）pp. 143~146に記載されるように、本来の物体表面の微小な凹凸に応じて近似多面体上の各部の法線に方向の変化を与え、かかる法線に応じた輝度を求めて表示するバンプマッピングが提案されている。

【0007】さらに、意匠上の効果のため、金属等の表面に規則的に細かなキズをつけ、見る方向により、輝きが異なるように加工した物、例えばステレオセットのつまみなどがある。この表示を行うため下記方法が考案されている。すなわち、見る方向により光の反射率および屈折率が変化することを考慮し、見る方向により前記反射率および屈折率の値を変更して表示するものである。この方法をテクスチャマッピングに適用した方法、すなわち、テクスチャデータの各画素毎に、上記変更のしかたを指定し、この指定に従ってテクスチャデータの各画素の値に変更を加えてマッピングを行い、表示する方法が特開昭63-213079号公報に開示されている。

## 【0008】

【発明が解決しようとする課題】ところで、特開平3-138779号公報に開示されている最も一般的なテクスチャマッピングでは、図形の表面の細部の紋様や材質感を表現することができ、静止した図形の表示の場合、ある程度の凹凸感を得ることができる。しかし、動く物体の表示や、静止した物体でも見る方向を変えるような表示、即ち、動画表示を行なう場合においては、近似多面体の表面に、即ち、平面上にテクスチャマッピングするため、その面の向きが変わっても、輝度計算により、輝度が均一に変化する以外には、本来の物体に存在する微小な凹凸による輝度の変化は起こらず、平板な表現となってしまう。

【0009】また、動画表示でバンプマッピングを適用するためには、表面の向きが変化する毎に微細な部分の輝度が法線の変化に応じて不均一に変化するようにしなければ、凹凸感が表現できない。しかし、このようにするためには、画素毎に輝度計算を行なう必要があり、現在の通常のハードウェアでは、処理に時間がかかりすぎて動画表示を行なうことは非常に困難である。

【0010】特開昭63-213079号公報に開示されているテクスチャデータの異方性を考慮する方法は、ミクロな凹凸によって、見る方向で見え方が変化することについては対応可能である。しかし、比較的大きな凹凸で、物体表面の紋様自体が変化する場合には対応できないため、凹凸感を表現することは難しく、質感表現の範囲にとどまる。上記のテクスチャマッピングやバンプマッピングについて、ドライビングシミュレータを例に挙げて更に説明する。

【0011】ドライビングシミュレータは運転操作の訓練等に用いられるものである。図12に示されるような表示装置8のスクリーン80上に表示される景観（物体

10

20

30

40

50

30を含む)を見ながら操作者が模擬的な運転操作機構を操作すると、その操作に応じてスクリーン上の景観が変化し、あたかも操作者が実際に自動車を運転しているかのような感覚を操作者に与えるものである。

【0012】図13は、自動車20が物体30の脇を通り過ぎる場合を示している。自動車20が位置P1から位置P4まで移動するとき、運転者から見える物体30の外観はその位置に応じて変化するはずである。したがって、自動車20の位置に応じて図12のスクリーン80上に表示される物体30の部分の紋様を変化させれば、ドライビングシミュレータを操作している操作者は、より高い現実感を持つことができる。

【0013】ところで、物体30としては、例えば、ビルディングの並びあるいはビルディングの壁面が考えられる。これらの表面には、図13に示したように凹凸が存在することが多い。ドライビングシミュレータの操作者の操作に応じて、スクリーン80上でこれらの凹凸が変化することが望ましい。しかしながら、このような場合、物体30を高速に表示するため、1つの矩形に対して、ビルディングの並びあるいはビルディングの壁面を表現する紋様をマッピングすることが多い。そのため、上述したテクスチャマッピングは、静止した図形の表示では凹凸感の表現に有効であるが、このような動画表示の場合には、平板な表現しか得られない。

【0014】また、これを補うために、上述したバンプマッピングが提案されているが、これは通常のハードウェアでは処理に時間がかかりすぎ、満足できる現実感のある動画表示を得るには大規模なハードウェアを必要とする。本発明の目的は、簡単なハードウェア構成で、動画表示においても、紋様や材質感とともに凹凸感を容易に表現することを可能にするテクスチャマッピング方法及び装置を提供することにある。

【0015】

【課題を解決するための手段】人間が凹凸感を感じる1つの大きな要素としては、物体が移動したり、物体に対する視点が移動した場合、物体の表面の細部の紋様の変化したり、輝度に変化することによるものと考えられる。ここで、図2により、凹凸感と視点位置あるいは物体の移動との関係をより具体的に説明する。図2(a)は凹凸がある面を示すものであって、100、101は表面が平坦で等幅、当間隔で配置された凸部である。ここでは、これら凸面100、101を明確に示すために、この面を若干左上方から見たものとしている。なお、この面は正面から照明されているものとする。

【0016】この面と光源との位置関係を固定として、この面を左上方の方向D1からみると、図2(b)に示すように、凸部100、101のあまり光が当たらない左側面100a、101aが見え、コーナを表わす縦方向に延びた線分の間隔は等間隔ではない紋様になる。一方、この面を正面上方の方向D2からみると、図2

(c)に示すように、縦方向に等間隔の線分が並んでいるように見える。また、この面を右上方の方向D3からみると、図2(d)に示すように、凸部100、101の右側面100b、101bが見え、縦方向の線分は等間隔ではない。このように、物体に対して視点位置が変わる毎に、凹凸に応じて紋様に変化する。これが、凹凸感を生む要素の1つと考えられる。

【0017】そこで、上記目的を達成するために、本発明は、かかる考察に基づき、次のようにして動画可能な凹凸感表示を実現するものである。即ち、テクスチャデータの基になる実際の物体の表面、あるいはコンピュータグラフィックスなどで作成した物体表面を見る方向を予めいくつか想定し、その方向毎に異なるテクスチャデータを設け、物体の表示時に、この物体の表面を見る方向に対応したテクスチャデータを1つ選択してマッピングする。

【0018】さらに、テクスチャデータの選択処理時において、物体の表面を実際に見る方向が予め想定した方向(以下、これを想定方向という)と一致しない場合には、これに近い複数の想定方向に対するテクスチャデータを選択し、これら選択されたテクスチャデータを実際に見る方向とこれら想定方向との偏差から補間処理を行ない、これによって得られたデータを実際に見る方向に対するテクスチャデータとして、表示される物体表面にマッピングしてもよい。

【0019】

【作用】上記のようにテクスチャマッピングを施すことにより、バンプマッピングを用いなくても、視点や表示図形が移動すると、それに応じてテクスチャデータが変化し、凹凸によるのと同じような紋様や輝度の変化を生じて凹凸感を得ることができる。さらに、複数のテクスチャの中から1つのテクスチャを選択するだけで、この処理はほとんど通常のテクスチャマッピングの処理と同等の時間で行なうことができる。

【0020】また、視点の移動や表示図形の移動により、実際に見る方向に厳密に該当するテクスチャデータがない場合でも、これに近い複数の想定方向に対するテクスチャデータを補間処理してこの実際に見る方向に対するテクスチャデータを得る。こうすれば、視線方向の変化により、マッピングする該当テクスチャが切り替わり、急激に紋様、材質感、凹凸感に変化することなく、自然な画像表示が可能となる。

【0021】

【実施例】以下、ドライビングシミュレータにおける本発明になるテクスチャマッピング表示を例にとり、本発明の実施例を図面を用いて説明する。図1は本発明によるテクスチャマッピング方法及び装置の一実施例を示すブロック図であって、1はCPU(中央処理ユニット)、2は主メモリ、3はシステムバス、4はグラフィックプロセッサ、41は幾何演算器、42は三角形発生

器、5はフレームメモリ、6はZバッファ、7はテキストメモリ、8はモニタ、22は入力装置である。

【0022】同図において、CPU1は、操作者（図示しない）によって操作される入力装置22からの入力に応じたテキストチャマッピングを施した三次元図形表示のために、図形データの管理やグラフィックプロセッサ4の起動などシステム全体の制御を行なう。主メモリ2は、図形データを記憶するほか、表示制御のためのプログラムを保持する。CPU1とグラフィックプロセッサ4とはシステムバス3を介して接続されている。グラフィックプロセッサ4はCPU1からの描画コマンドにより表示処理を実行するものであって、座標変換など幾何演算処理を行なう幾何演算器41とフレームメモリ5上にテキストチャマッピングされた三角形を描画する三角形発生器42とから構成されている。

【0023】フレームメモリ5上に作成された物体の画像がモニタ8に送られて表示される。Zバッファ6は、画面上他の図形によって隠される部分の表示を抑止する隠面消去のために用いられる。テキストメモリ7には、表示物体の表面を予め想定された各方向（即ち、想定方向）から見た場合のこれら各方向毎のテキストチャデータが記憶されている。入力装置22は、ドライビングシミュレータに対する操作者の操作、すなわち、模擬的なハンドル、ブレーキおよびアクセル操作をCPU1が解説可能なデータに変換して、CPU1あるいは主メモリ2に送り込む。

【0024】ここで、CPU1からグラフィックプロセッサ4に送られる三角形定義情報についての一般的な説明を図3～図5により行う。

【0025】図3は三角形の定義の方法を示すものであって、三角形は、一般に、図形を定義する座標系であるモデリング座標系上において、各頂点の座標値（ $x_j, y_j, z_j$ ）、各頂点での法線（ $nx_j, ny_j, nz_j$ ）、テキストチャ座標系での各頂点の座標値（ $s_j, t_j$ ）、面法線ベクトルN及びテキストチャアップベクトルUで定義される。但し、上記jは1, 2, 3のいずれかである。ここで、三角形の各頂点V1, V2, V3は夫々、図示するように、モデリング座標系上において、

V1 ( $x_1, y_1, z_1, nx_1, ny_1, nz_1, s_1, t_1$ )

V2 ( $x_2, y_2, z_2, nx_2, ny_2, nz_2, s_2, t_2$ )

V3 ( $x_3, y_3, z_3, nx_3, ny_3, nz_3, s_3, t_3$ )

と表わされ、これに面法線ベクトルN及びテキストチャアップベクトルUが設定されて三角形が定義される。ここで、テキストチャアップベクトルUは、マッピングされるテキストチャの上の方向を表わすベクトルである。なお、面法線ベクトルNは三角形の各頂点から、テキストチャアップベクトルUは頂点の座標およびテキストチャ座標から算出することもできる。

【0026】図4はテキストチャ座標系を示すものである。ここでは市松模様のテキストチャの例を示している。

図3に示した三角形にかかるテキストチャをマッピングする場合のテキストチャに対する三角形の位置関係も示している。テキストチャ座標系は直交するs, t座標軸の2次元座標系であり、このテキストチャ座標系でのテキストチャの定義域は、 $0 \leq s \leq 1.0, 0 \leq t \leq 1.0$ であって、正規化された2次元空間である。

【0027】テキストチャ座標系での三角形の位置は、図3で説明したこの三角形の各頂点のテキストチャ座標系での座標値（ $s_j, t_j$ ）で決まり（ここで、 $s_j$ はs座標軸方向の値、 $t_j$ はt座標軸方向の値であって、 $0 \leq s_j$ 及び $t_j \leq 1.0$ ）、三角形の各頂点の座標値V1 ( $s_1, t_1$ ), V2 ( $s_2, t_2$ ), V3 ( $s_3, t_3$ )によって、図4に示すように、テキストチャ座標系での三角形の位置が決まる。換言すると、この三角形には、その範囲内の市松模様のテキストチャがマッピングされることになる。なお、この三角形のテキストチャアップベクトルUの方向は、テキストチャ座標系のt座標軸の方向に一致する。

【0028】なお、ここでは、対象の表示図形を三角形とするが、一般に、表示物体は近似多面体で表現され、多角形は三角形に分割することが可能であるから、三角形に限定して説明しても一般性は失われない。表示図形が多角形の場合には、前処理としてこれを三角形に分割し、夫々の三角形について同様の処理を行なえばよい。また、テキストチャマッピング表示処理を実行するためには、表示物体の表面の材質の情報（即ち、光の反射係数や透明度などの属性情報）や使用テキストチャデータが必要となるが、これは既に指定されているものとする。

【0029】図5(a)から図5(d)は図2に示した凹凸がある物体30を図13の自動車20の位置P1～P4から見たときの物体表面の模様を示すものである。図5(a)から図5(d)での上部は物体表面の横断面図であり、下部はこの物体表面を水平面上にある中抜き矢印の方向から見たときの、それぞれのテキストチャTX1～TX4を示している。

【0030】図5(a)は自動車の位置P1から運転者が見た場合、すなわち、物体30の表面を左方約45度の方向から見た場合のテキストチャデータTX1を示し、2つの凸部100, 101の左側の側面100a, 101aがハッチングして示すように他の部分と異なって見えることになる。図5(b)は自動車位置P2から見た場合、すなわち、同じ物体30の表面を左方約25度の方向から見た場合のテキストチャデータTX2を示し、この場合も、図5(a)に比べて少ないが、2つの凸部100, 101の左側の側面100a, 101aがハッチングして示すように他の部分と異なって見えることになる。

【0031】図5(c)は自動車位置P3から見た場合、すなわち、同じ物体30の表面を正面から見た場合のテキストチャデータTX3を示し、2つの凸部100,

10:1の側面は見えない。図5(d)は自動車位置P4から見た場合、すなわち、同じ物体30の表面を右方向25度の方向から見た場合のテクスチャデータTX4を示し、2つの凸部100、101の右側の側面100b、101bがハッチングして示すように他の部分と異なって見えることになる。このように、同じ物体表面であっても、視線の方向が変化するとともにテクスチャも変化する。

【0032】さて、図3に示した三角形定義情報は、図1において、描画コマンドの形でCPU1からグラフィックプロセッサ4に送られる。グラフィックプロセッサ4では、幾何演算器41において、図6に示す処理が行なわれる。即ち、まず、三角形の頂点の座標、法線ベクトル、面法線ベクトルN及びテクスチャアップベクトルUを視点座標系に変換し(ステップ200)、表示範囲外の図形を切り取るクリッピングを行なう(ステップ201)。次に、光源の位置と強度、三角形の頂点の法線及び表示物体の表面の光の反射係数により、各頂点の輝度(即ち、シェーディング輝度)Rj、Gj、Bjを計算する(ステップ202)。なお、この輝度計算は、三次元グラフィックスでは一般に行われている事項であり、本発明に直接関係しないので詳細な説明は省略する。

【0033】次に、対象となるこの三角形を見る方向(視線の方向)に適したテクスチャデータをテクスチャメモリ7から選択する(ステップ203)。この選択処理の詳細を図7(a)および図7(b)を用いて説明する。なお、説明を理解しやすくするため、図5(a)～図5(d)に示すテクスチャデータTX1～TX4の1つが丸ごと選択され物体30の表面、すなわち、1つの矩形にマッピングされるものとする。

【0034】図7(a)において、テクスチャデータ管理テーブル10には、予め想定された方向から見たテクスチャデータTX1～TX4のテクスチャメモリ7上での格納位置を示すデータが保持管理されている。図7(b)に示すように、テクスチャメモリ7の格納領域はS、T座標系で表わされる。テクスチャデータTX1～TX4の各々は、このS、T座標系内における、各テクスチャデータの左下隅の座標(So, To)とその大きさ(ΔS, ΔT)を格納領域位置を示すデータとして、テクスチャデータ管理テーブル10内の該当行に格納されている。

【0035】但し、かかる領域は、実際には、テクスチャメモリ7内のアドレスの範囲を表わしている。ここでは、4種類のテクスチャデータTX1～TX4がテクスチャメモリ7内に格納されている。図7(a)における

$$S_j = S_o + s_j * \Delta S$$

$$T_j = T_o + t_j * \Delta T$$

で表わされる。但し、\*は乗算を示すものである。このようにして、テクスチャメモリ7のS、T座標系において、テクスチャに対する三角形の位置関係が決まる。

テクスチャデータ管理テーブル10において、例えば、テクスチャデータTX1の場合は、So, To, ΔSおよびΔTの値として、それぞれ、0、Tm, SmおよびTmが保持管理されている。

【0036】また、テクスチャメモリ7には、一種類の物体30の表面を様々な方向からみた紋様を表わすテクスチャTX1～TX4のみが記憶されていると仮定している。しかし、他の紋様のテクスチャデータを記憶しておいてもよいことは当然である。

【0037】かかるテクスチャデータ管理テーブル10でのデータは、図7(a)に示すようなレジスタ内のディスプレイメント11によりアクセスすることができる。このディスプレイメント11は視点座標系に変換された面法線Nを8ビットの精度で表現した情報Nxであって、テクスチャデータ管理テーブル10には、このディスプレイメント11の値をアドレスとして、各アドレスにディスプレイメント11の値に対応する上記のデータ(So, To, ΔS, ΔT)が保持管理されている。従って、このディスプレイメント11でもってテクスチャデータ管理テーブル10をアクセスすればよい。

【0038】この例では、ディスプレイメント11は“01011010”、すなわち、自動車位置P1での面法線のx成分Nxであって、この値により、ポイントテーブル12を参照して、テクスチャデータ管理テーブル10のエントリ番号(EPT)、すなわち2を得る。これから、EPTが2の場合に対応するテクスチャデータTX1の位置、すなわち、(So, To, ΔS, ΔT)の値(0, Tm, Sm, Tm)が得られる。ここで、Smはテクスチャメモリ7内のデータの仮想領域の横方向、すなわち、S軸方向の長さの半分を、Tmは同じく縦方向、すなわち、T軸方向の長さの半分の値を示す。

【0039】このようにして、物体30の表面を構成する各三角形の面の向き(即ち、視線の方向)も含めて、テクスチャメモリ7から所定のテクスチャデータ(ここではTX1)を読み出すことができる。

【0040】図6のステップ203では、さらに、テクスチャデータ管理テーブル10から得られた(So, To, ΔS, ΔT)の値からテクスチャメモリ7のS、T座標系での三角形の各頂点の座標位置を計算する。いま、j番目(但し、j=1, 2, 3)の頂点の0から1.0に正規化されたテクスチャ座標系(s, t)での座標位置を(sj, tj)とすると、テクスチャメモリ7上での座標系であるS、T座標系でのこの頂点の座標位置(Sj, Tj)は、

$$\dots\dots\dots (1)$$

$$\dots\dots\dots (2)$$

【0041】次に、図6において、上記三角形の各頂点の座標位置をモニタ8(図1)の画面に対応した座標系であるデバイス座標系の座標位置(Xj, Yj, Zj)に



変換し、三角形のかかる頂点の座標位置 ( $X_j, Y_j, Z_j$ )、テクスチャメモリ 7 の S、T 座標系での座標位置 ( $S_j, T_j$ ) 及び先に求めた輝度 ( $R_j, G_j, B_j$ ) を三角形発生器 42 (図 1) に送り、三角形を描画させる (ステップ 204)。なお、以上の処理を行なう幾何演算器 41 としては、マイクロプロセッサやメモリ、浮動小数点演算器、これらを制御するソフトウェアでもって構成することができる。

【0042】図 1 における三角形発生器 42 は、幾何演算器 41 からの上記パラメータを取り込み、フレームメモリ 5 上にテクスチャマッピングされた三角形を描画する。ここで、図 8 により、三角形発生器 42 の一具体例について説明する。但し、同図において、42a は座標・輝度発生部 42a、42b は輝度合成部、42c はフレームメモリ書込部である。

【0043】座標・輝度発生部 42a では、幾何演算器 41 から供給される三角形の各頂点のデバイス座標系での座標位置 ( $X_j, Y_j, Z_j$ )、テクスチャメモリ 7 の S、T 座標系での座標位置 ( $S_j, T_j$ ) 及び輝度 ( $R$

$$R_p = R_{sp} * R_{tp}$$

$$G_p = G_{sp} * G_{tp}$$

$$B_p = B_{sp} * B_{tp}$$

このようにして得られた輝度 ( $R_p, G_p, B_p$ ) は、フレームメモリ書込部 42c に送られる。

【0046】フレームメモリ書込部 42c では、座標・輝度発生部 42a から順次の画素のデバイス座標系での座標位置 ( $X_p, Y_p, Z_p$ ) が供給されるとともに、供給された座標位置 ( $X_p, Y_p, Z_p$ ) の画素に対する輝度 ( $R_p, G_p, B_p$ ) が輝度合成部 42b から供給され、この画素に対して Z バッファ 6 (図 1) に記憶されている Z 値と座標・輝度発生部 42a より送られた Zp 値とを比較し、処理中の画素の方が視点側、即ち、手前にある場合には、供給された上記の輝度 ( $R_p, G_p, B_p$ ) をフレームメモリ 5 (図 1) に書き込み、Z バッファ 6 の Z 値をこの Zp 値に更新する。処理中の画素の方が奥側の場合、これらの処理は行なわない。これにより、他の図形によって隠れる部分が表示されないようにする。

【0047】なお、座標・輝度発生部 42a の処理の詳細、即ち、三角形の内部画素の座標位置を頂点の座標位置からの補間処理によって得る方法については、フォーリ著、“コンピュータグラフィックス：プリンシパル アンド プラクティス 第 2 版”、アディソン・ウェスリ パブリッシング カンパニ インク刊、1990 年、第 18 章第 7 節 (Foley, "Computer Graphics : Principles and Practice second edition" Addison-Wesley Publishing Company Inc 発行 (1990), Chapter 18.7) に開示されている。

【0048】このようにして、図 12 に示すようにモニタ 8 のスクリーン 80 上には、45 度左斜めから見た紋

j,  $G_j, B_j$ ) が補間処理されることにより、この三角形内部の各画素のデバイス座標系での座標位置 ( $X_p, Y_p, Z_p$ )、S、T 座標系での座標位置 ( $S_p, T_p$ ) 及びシェーディング輝度 ( $R_{sp}, G_{sp}, B_{sp}$ ) が順次生成される (但し、これらの座標位置やシェーディング輝度は、三角形の各頂点のものも含む)。座標位置 ( $X_p, Y_p, Z_p$ ) はフレームメモリ書込部 42c に、座標位置 ( $S_p, T_p$ ) はテクスチャメモリ 7 (図 1) に、シェーディング輝度 ( $R_{sp}, G_{sp}, B_{sp}$ ) は輝度合成部 42b に夫々送られる。

【0044】座標・輝度発生部 42a からの上記の座標位置 ( $S_p, T_p$ ) により、テクスチャメモリ 7 からテクスチャ輝度 ( $R_{tp}, G_{tp}, B_{tp}$ ) が読み出され、輝度合成部 42b に供給される。この輝度合成部 42b では、シェーディング輝度 ( $R_{sp}, G_{sp}, B_{sp}$ ) とテクスチャ輝度 ( $R_{tp}, G_{tp}, B_{tp}$ ) とが下記の式で示されるように合成され、画素の輝度 ( $R_p, G_p, B_p$ ) が得られる。

【0045】

$$\dots\dots\dots (3)$$

$$\dots\dots\dots (4)$$

$$\dots\dots\dots (5)$$

様、すなわち、テクスチャデータ TX1 の表示が得られる。ここで注意しておきたいのは、モニタ 8 上の物体 30 は平板状の領域であって、この上にマッピングしたテクスチャの紋様によって凹凸感を出していることである。

【0049】ところで、以上の説明では、テクスチャメモリ 7 からは、予め準備されているテクスチャデータのうちの視点方向に最も近い想定方向に対するテクスチャデータが選択される。しかし、このようにすると、視点の移動や物体移動により、あるテクスチャデータから他のテクスチャデータに急激に切り替わることになり、テクスチャ輝度に不連続な部分が生じて不自然な表示画像が発生する。

【0050】これを防止するためには、視点の方向に最も近い複数個の想定方向に対するテクスチャデータを選択し、これらテクスチャデータを視点方向とこれら複数の想定方向との偏差に応じて補間処理し、視点方向に対するテクスチャデータを求めるようにすればよい。これにより、表示物体の表面での輝度が連続したものとなり、良好な表示画像が得られる。以下、このための処理について説明する。

【0051】いま、視点方向に最も近い 2 つの想定方向に対するテクスチャデータのテクスチャ輝度を ( $R_{t1}, G_{t1}, B_{t1}$ )、( $R_{t2}, G_{t2}, B_{t2}$ ) とし、これら 2 つの想定方向の角度差に対するこれら 2 つの想定方向のうち的一方と視点方向の角度差との比を補間係数を  $\alpha$  とすると、視点方向に対するテクスチャデータのテクスチャ輝度 ( $R_t, G_t, B_t$ ) は、

$$R_t = \alpha R_{t1} + (1 - \alpha) R_{t2}$$

$$G_t = \alpha G_{t1} + (1 - \alpha) G_{t2}$$

$$B_t = \alpha B_{t1} + (1 - \alpha) B_{t2}$$

として求められる。

【0052】このようにすると、視点や表示物体が移動した場合、表示物体の表面上の紋様が急激に変化することはない。これらの処理は、グラフィックプロセッサ4の幾何演算器41と三角形発生器42の処理の一部を変更することによって可能である。以下、この処理について説明する。

【0053】幾何演算器41は、図6におけるステップ203において、視点方向に最も近い想定方向に対する複数のテクスチャデータを求めるのであるが、これは、図9に示すように、図7に示した方法にポイントテーブル12に、ディスプレイメント11に対応する最も近い2つのテクスチャデータを示すポイントと補間係数 $\alpha$ とを保持することにより実現する。即ち、まず、ディスプレイメント11でポイントテーブル12をアクセスし、ディスプレイメント11で設定されている視点の方向に最も近い2つの想定方向に対するテクスチャデータ管理テーブル10のエントリへのポイントP1、P2と補間係数 $\alpha$ とを求める。このポイントP1、P2によってテクスチャデータ管理テーブル10のアクセスが行なわれ、視点の方向に最も近いこれら2つの想定方向夫々に対するテクスチャデータが得られる。

【0054】図9の8ビットで表現されるディスプレイメント11が“01001000”、すなわち、図13に示した自動車位置P1とP2との中間の場合、ポイントテーブル12からテクスチャデータ管理テーブル10のエントリ番号（この場合“3”と“2”）と補間係数 $\alpha$ （この場合“0.5”）とが得られる。さらに、テクスチャデータ管理テーブル10から2つのテクスチャデータ（この場合TX1とTX2）が選択されることになる。これらテクスチャデータから、上記のように、テクスチャメモリ7のS、T座標系での座標位置（S1、T1）及び（S2、T2）を求めることができる。

【0055】1つの三角形の頂点のデバイス座標系での座標位置、シェーディング輝度に加えて、上記処理で求めた2組のS、T座標系での座標位置と補間係数 $\alpha$ が三角形発生器42に送られる。図6のステップ204では、三角形発生器42により、次のような処理が行なわれる。

【0056】即ち、図8において、座標・輝度発生部42aは、供給された上記のパラメータから、1つの三角形に対して、三角形内部の各画素のデバイス座標系での座標位置、S、T座標系での座標位置、シェーディング輝度を生成し、デバイス座標系での座標位置をフレームメモリ書き込み部42cに、S、T座標系での座標位置をテクスチャメモリ7（図1）に、シェーディング輝度と補間係数 $\alpha$ を輝度合成部42bに夫々供給する。輝度合

$$\dots\dots\dots (6)$$

$$\dots\dots\dots (7)$$

$$\dots\dots\dots (8)$$

成部42bでは、S、T座標系での座標位置によってテクスチャメモリ7から読み出される2組のテクスチャ輝度を取り込み、これら2組のテクスチャ輝度と補間係数 $\alpha$ とで上記の式（6）～（8）の補間処理演算を行なって視線方向に対するテクスチャデータのテクスチャ輝度を求める。

10 【0057】こうして得られたテクスチャ輝度は、上記の式（3）～（5）に従ってシェーディング輝度と合成され、以下、先に説明したように処理される。このようにして、図10に示したように、テクスチャデータTX1とテクスチャデータTX2とを合成し、合成されたテクスチャデータTXiをマッピングしても、近似的な表示を得ることができる。これによると、凸部の側面の部分が、テクスチャデータTX2での凸部のハッチングで示す側面部分の外側に、テクスチャデータTX1で広くなっている部分が輝度が弱められて描かれ、本来の凸部の側面部分の幅のように見えることになる。

20 【0058】いま、図13で示した視点位置の変化は、視点座標系での表示物体の面法線のX成分のみで表現可能であり、従って、ディスプレイメント11として、面法線のX成分を用いた。視点位置の変化の方向が特定出来ない場合は、図11（a）に示すように、ディスプレイメント11として、面法線NおよびテクスチャアップベクトルUのXYZ成分を所定の精度のビット数で表現したものとし、ポイントテーブル12にて、テクスチャデータ管理テーブル10のエントリ番号EPTに変換するようにすればよい。また、図11（b）に示したように、横Smax、縦Tmaxの大きさのテクスチャメモリ7上においてテクスチャデータは左下隅の位置（Si、Ti）と横および縦の長さ（ $\Delta Si$ 、 $\Delta Ti$ ）を指定することにより、先に述べた4種類のテクスチャデータだけでなく、これ以上の種類のテクスチャデータを保持することが出来る。

【0059】

【発明の効果】以上説明したように、本発明によれば、表示物体の表面の向きにより、その向きに該当するテクスチャデータを従来方法でマッピングするだけですむ。そのため、簡単なハードウェアを用いても従来のテクスチャマッピングとほぼ同等の処理時間で処理することができる。このために、動画表示時、表示物体の微細な部分の輝度が変化し、凹凸感を表現することができる。

【0060】さらに、視線方向に最も近い複数の想定方向に対するテクスチャデータを補間処理によって得て、その値をマッピングすることができる。したがって、視点や表示物体が移動した場合でも、自然に滑らかに紋様が変化し、急激に紋様が変わることはない。自然な動画表示が可能になる。

【図面の簡単な説明】

【図 1】 本発明によるテクスチャマッピング方法及び装置の一実施例を示すブロック図である。

【図 2】 凹凸面の視点位置の変化による見え方の変化の例を示す図である。

【図 3】 三角形の定義方法の説明図である。

【図 4】 テクスチャ座標系での三角形の位置を説明するための図である。

【図 5】 テクスチャデータの一例を示す図である。

【図 6】 図 1 における幾何演算器の処理動作を示すフローチャートである。

【図 7】 図 6 におけるステップ 203 の処理を説明するための図である。

【図 8】 図 1 における三角形発生器の一具体例を示すブロック図である。

【図 9】 補間処理を行なう場合の図 6 におけるステップ 203 の処理を説明するための図である。

【図 10】 テクスチャデータの補間の一具体例を示す図である。

【図 11】 視点方向が複雑に変化する場合の図 6 のステップ 203 の処理を説明するための図である。

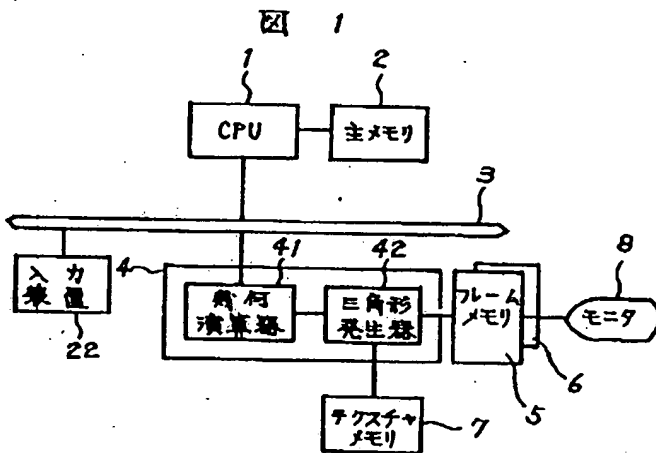
【図 12】 ドライビングシミュレータの表示画面の例を示す図である。

【図 13】 視線方向の変化に応じた物体の表面の変化を説明するための図。

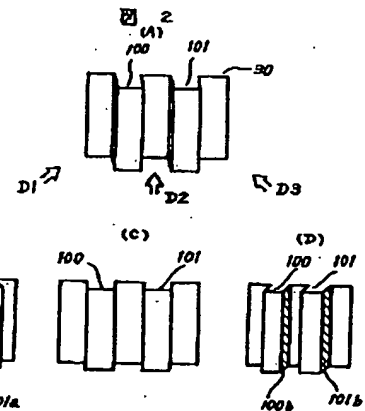
【符号の説明】

1…CPU、2…主メモリ、3…システムバス、4…グラフィックプロセッサ、5…フレームメモリ、6…Zバッファ、7…テクスチャメモリ、8…モニタ、22…入力装置、10…テクスチャデータ管理テーブル、11…ディスプレイメント、12…ポインタテーブル、42a…座標・輝度発生部、42b…輝度合成部、42c…フレームメモリ書込部。

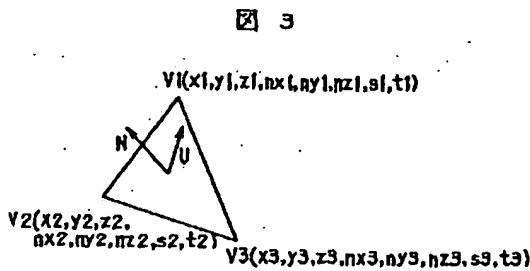
【図 1】



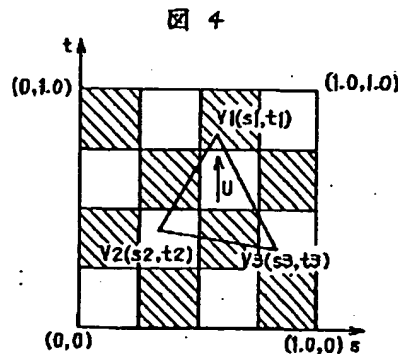
【図 2】



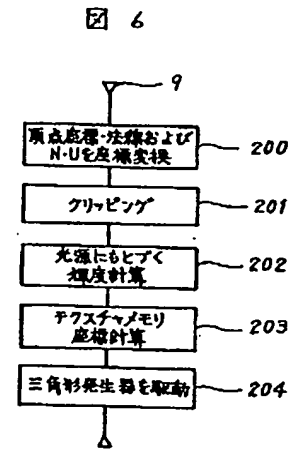
【図 3】



【図 4】



【図 6】



४८

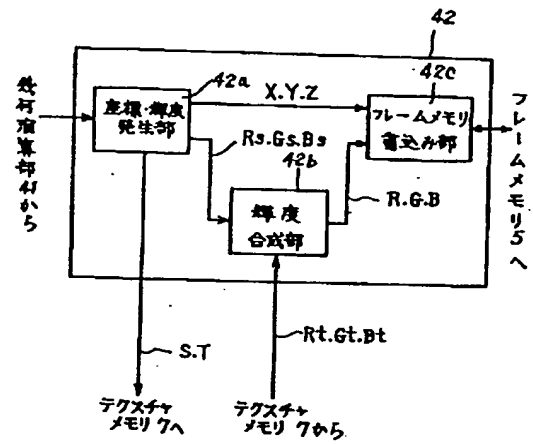
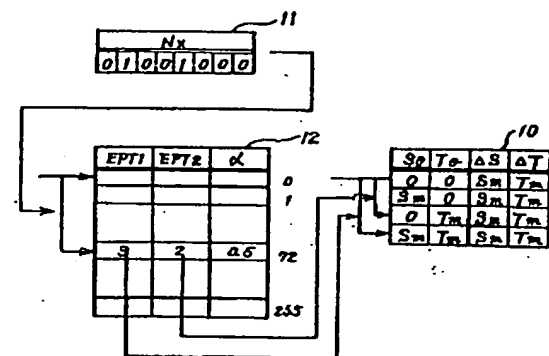
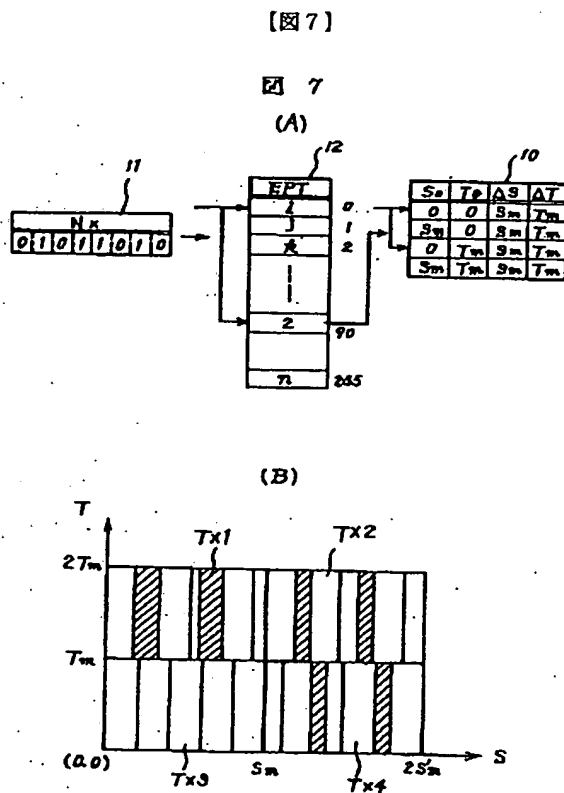
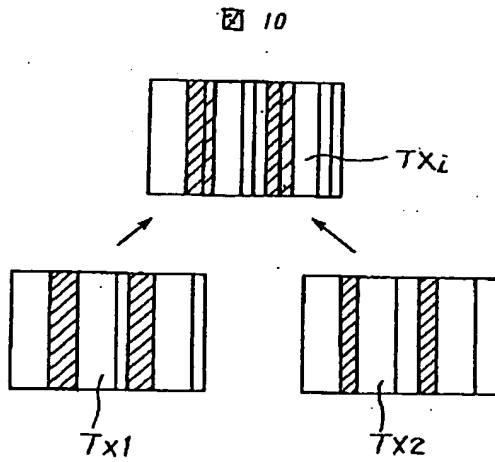


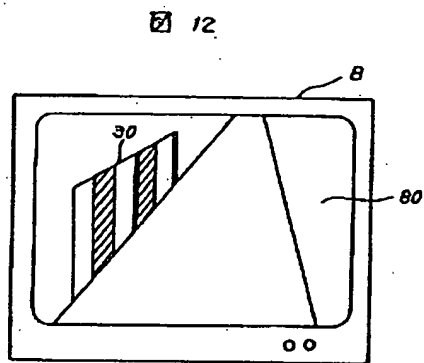
图 9



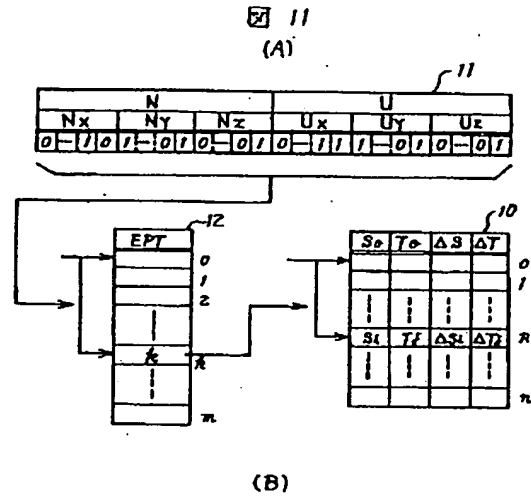
【図10】



【図12】



【図11】



【図13】

